

#8  
WM  
4/8/03

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

First Named Applicant: Choy

Serial No.: 09/614,369

Filed: July 12, 2000

For: **SYSTEM AND METHOD FOR ENSURING  
REFERENTIAL INTEGRITY FOR  
HETEROGENEOUSLY SCOPED REFERENCES IN  
AN INFORMATION MANAGEMENT SYSTEM**

Art Unit: 2172

Examiner: Woo

AM9-99-0209

March 27, 2003  
750 B STREET, Suite 3120  
San Diego, CA 92101

**RECEIVED**

APR 03 2003

Technology Center 2100

APPEAL BRIEF

Commissioner of Patents and Trademarks  
Washington, DC 20231

Dear Sir:

This appeal brief is submitted under 35 U.S.C. §134. This appeal is further to Appellant's Notice of Appeal filed herewith.

Table of Contents

<u>Section</u>	<u>Title</u>	<u>Page</u>
(1)	Real Party in Interest .....	1
(2)	Related Appeals/Interferences .....	2
(3)	Status of Claims .....	2
(4)	Status of Amendments .....	2
(5)	Summary of Invention .....	2
(6)	Issues .....	2
(7)	Grouping of Claims .....	3
(8)	Argument .....	3
App.A	Appealed Claims	

(1) **Real Party in Interest**

The real party in interest is IBM Corp.

04/08/2003 WHEREIN 00000002 090441 09614369  
01 FC:1402 320.00 CH

**(2) Related Appeals/Interferences**

No other appeals or interferences exist which relate to the present application or appeal.

**(3) Status of Claims**

Claims 1-14 are pending and finally rejected.

**(4) Status of Amendments**

An amendment cancelling Claims 15-19 is submitted herewith.

**(5) Summary of Invention**

As set forth in, e.g., Claim 1, the invention is a system for providing referential integrity for heterogeneous links that includes an RDBMS providing referential integrity for homogeneous links, and a software layer on top of the RDBMS. The software layer causes the RDBMS to provide referential integrity for heterogeneous links, in part by the software layer maintaining at least one data structure useful in ensuring referential integrity.

Claim 5 recites a method for preventing dangling pointers in heterogeneously scoped links that includes providing a heterogeneously scoped link (HSL) table in a non-RDBMS element that communicates with an RDBMS. An RDBMS table has a heterogeneously scoped link column and the HSL table is associated with the heterogeneously scoped link column. The method includes accessing the HSL table to ensure referential integrity in an RDBMS.

**(6) Issues**

(a) Whether Claims 1-4 are unpatentable under 35 U.S.C. §103 as being obvious in light of Vijaykumar in view of Sarkar.

(b) Whether Claims 5-14 are unpatentable under 35 U.S.C. §103 as being obvious in light of Sarkar.

**(7) Grouping of Claims**

Owing to the different bases for rejection, Claims 1-4 must be grouped apart from Claims 5-14.

**(8a) Argument**

Claims 1-4 have been rejected as being unpatentable over Vijaykumar, which as admitted in the Office Action fails to disclose heterogeneous links, much less a software layer on top of an RDBMS for ensuring integrity of such links, in view of Sarkar, used as a teaching of a software layer on top of an RDBMS for providing referential integrity for heterogeneous links.

On the contrary, neither Vijaykumar nor Sarkar teach or suggest a table or other data structure in a software layer apart from an RDBMS for maintaining referential integrity within the RDBMS. Vijaykumar provides a preferred user interface for initially defining integrity constraints just as is required in conventional RDBMS, but the structures for actually maintaining the integrity are "internal" to the RDBMS (either a table or RDBMS ".val" file is used, col. 3, lines 41-47.) There is thus no data structure apart from RDBMS data structures in Vijaykumar that is useful for maintaining (as opposed to defining) referential integrity, in contrast to Claim 1.

Sarkar offers no better teaching. In fact, sorting through the volume of text in Sarkar cited in the Office Action and focussing on the relevant issue, Sarkar teaches the very opposite of what is claimed at col. 10, lines 18-20. Specifically, Sarkar explicitly states that the table R', which is part of RDBMS schema 1,

maintains the referential integrity of the system. Sarkar explains that local maintenance of dependency information is necessary for consistency, col. 10, lines 26-30. The non-RDBMS software layers in Sarkar evidently only facilitate information exchange, col. 10, lines 20-22, without providing any actual integrity management, in contrast to Claim 1. Indeed, according to Sarkar, "[t]his invention extends these prior arts by using Uniform Resource Locators (URLs) *in relational databases* to reference objects...", col. 5, lines 27-30 (emphasis mine). In fact, according to Sarkar, "[a] uniform paradigm for multi-tier client/server *without a middle tier application server* is presented", col. 5, lines 45-48 (emphasis mine).

In other words, Sarkar fails to recognize the desirability of maintaining referential integrity for heterogenous links without modifying the underlying RDBMS, in marked contrast to the recognition in the present specification on page 2 ("One way to provide heterogenous linking referential integrity is to modify the RDBMS. This is not a trivial task. Accordingly, the present invention is directed to a system and method for providing heterogenous linking referential integrity without modifying the RDBMS..."). Not surprisingly, Sarkar sees no problem with putting its table R' in an RDBMS, or with using URLs within an RDBMS, instead of using a software layer on top of the RDBMS to maintain a data structure for ensuring referential integrity as set forth in Claim 1. Accordingly, the references, even if combined as proposed, would not arrive at Claim 1, and in fact Sarkar in particular expressly teaches away from it.

Moreover, the stated motivation for the proposed combination is somewhat opaque, but regardless, the motivation does not appear to come from the prior art, as is otherwise required by MPEP §2143.01. Instead, the relied-upon motivation to combine the references on page 3 of the Office Action appears to be a regurgitation of Applicant's own teachings. For this further reason, a prima facie case of obviousness has not been properly established under the MPEP.

CASE NO.: AM9-99-0209  
Serial No.: 09/614,369  
March 27, 2003  
Page 5

PATENT  
Filed: July 12, 2000

(8b) Argument

Claims 5-14 have been rejected as being obvious over Sarkar. Both independent claims in this group (5 and 11) require a table apart from an RDBMS that is used for maintaining referential integrity within the RDBMS (Claim 5) or that is used for preventing dangling pointers (Claim 11). For the reasons set forth above, Sarkar not only fails to suggest this, it expressly teaches away from it.

Respectfully submitted,



---

John L. Rogitz  
Registration No. 33,549  
Attorney of Record  
750 B Street, Suite 3120  
San Diego, CA 92101  
Telephone: (619) 338-8075

JLR:jg

## APPENDIX A - CLAIMS

1. A system for providing referential integrity for heterogenous links, comprising:  
an RDBMS providing referential integrity for homogenous links; and  
a software layer on top of the RDBMS for causing the RDBMS to provide referential integrity for heterogenous links, the software layer maintaining at least one data structure useful in ensuring referential integrity.
2. The system of Claim 1, wherein the software layer maintains at least one table.
3. The system of Claim 2, wherein the table is accessed upon an attempted deletion or updating of a tuple referenced by a link, and the attempted deletion or updating is selectively disallowed based on the table.
4. The system of Claim 2, wherein the software layer includes at least one stored procedure accessible by an application to insert, update, or delete a tuple while ensuring referential integrity in heterogenous links associated with the tuple.
5. A computer-implemented method for preventing dangling pointers in heterogeneously scoped links, comprising the acts of:  
providing at least one heterogeneously scoped link (HSL) table in a non-RDBMS element communicating with at least one RDBMS, at least one table having a heterogeneously scoped link column, the HSL table being associated with the heterogeneously scoped link column; and  
accessing the HSL table to ensure referential integrity in an RDBMS.
6. The method of Claim 5, wherein the HSL table is accessed when a link attribute is sought to be changed.
7. The method of Claim 5, wherein the HSL table is accessed when a tuple is sought to be changed or deleted.
8. The method of Claim 5, wherein the HSL table is established by an RI table.
9. The method of Claim 5, further comprising providing at least one trigger useful in selectively disallowing operations.
10. The method of Claim 5, further comprising providing at least one stored procedure accessible by an application to insert, update, or delete a tuple while ensuring referential integrity in heterogenous links associated with the tuple.

11. A computer program product including computer usable code means programmed with logic for ensuring referential integrity in an RDBMS having at least one table with at least one column of heterogeneously scoped links, the program product comprising:

computer readable code means for maintaining a table in a software layer not part of the RDBMS; and

computer readable code means for using the table to ensure that operations on tuples in the RDBMS do not result in a heterogeneously scoped link pointing to no tuple.

12. The computer program product of Claim 11, further comprising:

computer readable code means for establishing at least one trigger useful in cooperation with the table for selectively disallowing operations.

13. The computer program product of Claim 12, further comprising computer readable code means for establishing at least one of: a delete trigger, and an update trigger.

14. The computer program product of Claim 11, further comprising computer readable code means for inserting, updating, or deleting a tuple while ensuring referential integrity in heterogenous links associated with the tuple.